MultiBit Wallet Improvements

Introduction

This document describes proposed changes to the current MultiBit UI to improve the capability of wallets.

Supported wallet types

There are the following types of wallets to support:

- Unencrypted wallets with random keys. These are currently supported.
- Encrypted wallets with random keys. The MultiBit v0.5 branch supports these however I want to change the encrypted wallet format to support wallet recovery in case of losing your password. This is extra work.
- Wallets conforming to the Hierarchical Deterministic (HD) specification described here: https://en.bitcoin.it/wiki/BIP_0032 . These are not supported yet.
- The Slush/ Stick bitcoin hardware wallet also uses HD wallets and I want to support this device. From MultiBit's point of view these are watch only wallets (i.e. no private keys present). Watch only wallets are currently not supported in MultiBit.

This document describes the user interface changes first and then the consequent coding/ model changes towards the end.

Explanation of the wallet recovery page

It is fairly common for users to forget their passwords. To enable them to recover their wallets in this case I propose adding a **wallet master key** to the bitcoinj wallet format.

Currently encryption is done as follows:

1. User enters a password.
2. AES key is generated from password using scrypt.
3. Wallet keys are encrypted with the AES key from step 2.

After a suggestion from Alan Reiner, I propose changing this to:

1. A randomly generated wallet master key is generated. This is an AES key.
2. Wallet keys are encrypted with the wallet master key.
3. User enters a password.
4. AES key is generated from the user password using scrypt.

5. The wallet master key is encrypted with the key from step 4 and stored in the wallet.

As the wallet master key never changes for a wallet the user can print it out on a sheet of paper and put it somewhere safe. If they forget their password, they can use it to recover the information in their wallet.

Overview of wallet functionality

To support the various wallet types, the various screens need creating or modifying.

- A ‘New Wallet’ screen’ in which the user can choose what sort of wallet to create, specify the necessary details and then create the wallet. This will be accessed from the ‘New’ button in the wallet side panel and from the “File | New Wallet” menu option.
- Menu options in the ‘File’ menu where the user can:
    - ‘Add Password’. Add a password to an existing, non-password protected wallet.
    - ‘Change Password’. Change the password for a password protected wallet
    - ‘Remove Password’. Remove the password from a password protected wallet
- Menu options in the ‘Tools’ menu where the user can:
    - ‘Print Wallet Recovery Page’. The user can generate and print out a sheet of paper with their wallet master key on.
      FUTURE: It will also have the seed/ chain code and/ or mnemonic phrase for their HD wallet.
    - ‘Recover Wallet’. The user can enter the wallet master key for a wallet and set a new password.
- When sending bitcoin, the password for the wallet must be specified if required. This will be on the ‘Send Bitcoin Confirm dialog”.
- The import and export private keys options need to be aware of if the wallet is password protected or not. When exporting private keys the password for the wallet must be specified. Also when importing keys the keys that are added to the wallet need to be encrypted using the wallet master key (decrypted by the wallet password).

Slush/ Stick hardware wallet support

- Wallet type is a watch only HD wallet.
- When user plugs in device either:
    - create new wallet in user data area

  or
    - open existing matching wallet
- Name the wallet for a particular device as <unique identifier>.wallet
- Note on wallet naming from Alan: In the current incarnation of Armory wallets, I had used a combination of the root address and the first address after it to create a mostly-unique

6-byte ID for the wallet. This was so that a given combination of root seed and chaining algorithm would have unique IDs. Chaining algorithm doesn't have to be part of it with standardized BIP 32 (maybe just food for thought)... but I think it is appropriate to add an identifier scheme which would solve your problem. Jim: this would be good to have in BIP32 as then the same device would have the same wallet identifier on different clients.

‘New Wallet’ screen

The ‘New Wallet’ screen will be in the same style as the screens such as ‘Import private keys’ and look like:

Wallet type

Choose the type of wallet to create:

<combo box with options> Random key << default
Deterministic << *FUTURE*
Watch only << *FUTURE*

Next section only appears if watch only wallet type is selected

Wallet to watch

Select the wallet you want to watch in the wallet panel on the left.

Description <Description of wallet chosen>
Filename <Filename of wallet chosen>

Next section only appears if deterministic wallet type is selected

Mnemonic pass phrase

If you want to recreate an existing deterministic wallet, enter the mnemonic pass phrase

Pass phrase <Mnemonic pass phrase field>

Need a date entry field to get date for replay - slider ?

Output file

Choose where you would like to save your wallet.

Filename <File location>
<Choose wallet file> button

Description <Wallet description entry field>

In the next section the ‘Do not password protect’ is hardwired for watch only wallets (as they have no private keys it is pointless to encrypt them).

Password

<radio> Password protect wallet file:

Password: <password entry field>

Repeat password: <repeat password entry field>

<radio> Do not password protect wallet file

Keep your password safe.
You need this password to spend your bitcoin.

Wallet recovery page

Your wallet recovery page will appear after you press the ‘Create new wallet’ button.
Print this out and keep it safe.
You can use it to recover your wallet if you lose your password.

<Create new wallet> button
<<Status line used to report back status of wallet creation>>
<<Status line used to report on encrypted private key backup file, if applicable>>
(?) << *The context sensitive help icon*

‘Add Password’ screen

Wallet

Description: <description of wallet>
Filename: <filename of wallet>

Password

Password: <password field>
Repeat password: <repeat password field>

Keep your password safe
You need this password to spend your bitcoin.

Wallet recovery page

Your wallet recovery page will appear after you press the ‘Add password to wallet’ button.
Print this out and keep it safe.
You can use it to recover your wallet if you lose your password.

<Add password to wallet> button
<<Status/ feedback line>>

<<Feedback line giving location of encrypted private key backup file>>
(?) << *The context sensitive help icon*

‘Change Password’ screen

Current Password

Password: <password entry field>

New Password

Password: <password field>
Repeat password: <repeat password field>

<Change password> button
<<Status/ feedback line>>
<<Feedback line giving location of encrypted private key backup file>>
(?) << *The context sensitive help icon*

‘Remove Password’ screen

Password

To remove the password protection for this wallet, enter the existing password.

Password: <password entry field>

<Remove password> button
<<Status/ feedback line>>
(?) << *The context sensitive help icon*

‘Print Wallet Recovery Page’ screen

Instructions

1. Enter the wallet password and press the ‘Print wallet recovery page’ button.
2. Your wallet recovery page will appear.
3. Print this off and keep it safe.

You can use it to recover your wallet if you lose your password.

Wallet

Description: <description of wallet>
Filename: <filename of wallet>

Wallet password: <wallet password entry field>

<Print wallet recovery page> << Button
<< Status line>>
(?) << Context sensitive help icon

‘Recover Wallet’ screen

Wallet

Select the wallet you want to recover in the wallet panel on the left.

Description: <description of wallet>
Filename: <filename of wallet>

Wallet master key

Enter the ‘Wallet master key’ from your wallet recovery page.

Wallet master key < Wallet master key entry field >

<< Status line indicating whether the wallet master key is valid = can decrypt first encrypted key of wallet accurately >>

New Password

Password: <password field>
Repeat password: <repeat password field>

<Recover wallet> << button
<<Status line indicating status of wallet recover>>
(?) << Context sensitive help

Layout of the ‘Wallet Recovery Page’

MultiBit
Wallet Recovery Page

Wallet

Description: <description of wallet>
Filename: <filename of wallet>
Creation date: <date wallet was created>

Wallet master key

The 32 bytes of the wallet master key, shown in hex

FUTURE Deterministic key data

Seed < the seed for the HD wallet in hex>
Chain code < the chain code for the HD wallet in hex>

and/ or
Mnemonic pass phrase <The mnemonic passphrase for the wallet>

Anyone who can read this file can spend your bitcoin

<Print> <Cancel> << buttons on the dialog screen, not printed.

Print on a single page, smaller than both A4 and the standard American page size so that it will print on one sheet.

Coding/ Model notes

- Need new wallet type: protobuf.4 (with wallet master key - optional bytes in Wallet message)
- There needs to be a standard for creation of mnemonic pass phrases from the HD seed and chain code - not specified in BIP 32. Propose using the Electrum mnemonic algorithm specified in https://github.com/spesmilo/electrum/blob/master/lib/mnemonic.py
- Would be good to have a standard way of creating wallet identifiers for HD wallets so the same Slush/ Stick device has the same name on different clients.
- Watch only versions of unencrypted/ random key wallets cannot create new private keys - disable button on ‘Request bitcoin’ tab.
- Watch only versions of HD wallets CAN create new receiving addresses.
- Watch only wallets need wallet auto-sync when opened as they will always be out of date. We have the last seen block in the wallet already. Use “wallet busy” to make them unavailable whilst syncing. Will probably have to have a “syncing” jobs queue to mange the threads.
- Wallet auto-sync also means there can be a ‘Close wallet’ operation to remove a wallet from the list of wallets.